Python Programmierung 2

13.11.2023 bis 22.01.2024

Durchführung: 23H__PYT2_22NI Lehrperson: Walter Rothlin

Block	Seq	Dauer	Art	Inhalt	Ziele	Methode	Hausaufgabe
01	1	4.00L	PU	 * Repetition class(), methoden, instance-variablen, properties * private / public * Statische Methoden * docStrings * Vererbung * Reflaction (Zugriff auf docStrings, dynamisch Function-Call) * Enumerations (ChatJPT: Good morning. Pls show me some examples for enums in Python) * Sub-Classes (e.g. Sense_Hat) 	Vertiefen des Klassen-Konzept in Python	Problem-Based	Code-Refactoring
02	1	4.00L	PU	* Requirements überarbeiten * Interface (Abwärtskompatible) überarbeite (Cleancode) * Public/private/Properties/setter/getter * Naming * ENUM für Log-Level * docStrings / API Doc Refactoring Weather Application (Usage of Logger-Klasse)	Refactoring Logger-Klasse	Problem-Based	Code-Refactoring
03	1	1.20L	PU	Leistungskontrolle 1 (Design and Implement a Class)			
03	2	2.80L	PU	 * Unterschied Packages / Module * Dokumentation von Modulen * Namespaces * Selektiver Import * Abfangen von Exceptions: Try / except / else / finally * Exception werfen: Raise * Exception-Handling: lösen/werfen * Eigene Exceptions 	 * Umgang mit Modulen und Packages * Exception Handling in Python 	Problem-Based	Code-Refactoring
04	1	4.00L	PU	* Threads kreieren (fork) / stoppen * Kommunikation zwischen Threads * Thread-Synchronisation * Locks * I/O Interrupts * Time-Interrupts	Multithreadfing in Python	Problem-Based	Code-Refactoring
05	1	1.20L	PU	Leistungskontrolle 2 (Design and Implement a Sub-Class)		2 11 -	
05	2	2.80L	PU	* GPIO (LED-Steuerung) * PiPlates * Relais-Karte * Analog/digital I/O Mutteruhr für SBB-Uhr Selecta-Automat	Hardware ansteuern	Problem-Based	Code-Refactoring
06	1	4.00L	PU	* Template-Mechanism (Jinja2)	Template Mechanismus	Problem-Based	Code-Refactoring
				* ChatJPT: Give me an example for JINJA Templates?			



Block Seq Dauer Art Inhalt Ziele Methode Hausaufgabe * ChatJPT: Can I also implement loops and if-then-else in templates? * Hello-Word with FLASK Web-Anwendung mit FLASK 07 4.00L DL Problem-Based Code-Refactoring * Request / Responses * Static HTML-content * Mimetype und JSON responses * Parameter Übergabe (get/put) * Endpoints Modullernzielkontrolle (MLZ) Ein lauffähige 80 4.00L * Eine lauffähige Applikation nach Vorgaben unter Zeitdruck Applikation nach Vorgaben unter implementieren. Zeitdruck implementieren. 4.00L Einzelne Fachgespäche über MLZ Einzelgespräche PU Selbständiges Arbeiten an: Einzelarbeit * XML processing in Python (XPath, Schemas, XSLT) * Excel read/write access * RegEx

Bildungszentrum Uster							
Höhere	Fachs	chule					
Uster							
Tankaile Informs	atile Winteelead	4 F/M					

Block	Seq	Dauer	Art	Inhalt	Ziele	Methode	Hausaufgabe